A Robotics Multidisciplinary Project Action Group

Jerry B. Weinberg* and Gary R. Mayer**
Department of Computer Science * School of Engineering
Southern Illinois University – Edwardsville
Edwardsville, IL 62026-1656
*jweinbe@siue.edu, ** malekith@alum.wpi.edu

Abstract

Robotic projects encompass the rich nature of integrated systems that includes computational, mechanical, and electrical components. Recent development of robot platforms has made robotics development accessible to anyone regardless of their background knowledge in these areas. Yet using robotics projects effectively as a learning experience does require some understanding of these areas. To address this we have formed a multidisciplinary group of instructors that provides a basis for sharing expertise and designing educational experiences.

Introduction

Robots are the quintessential example of integrated system engineering. They combine interacting mechanical, electrical, and computational components. Their multidisciplinary nature has in the past relegated the study of robotics to larger research universities and private industrial research groups whose members had the full range of prerequisite knowledge to engineer such complex systems. Preconstructed industrial robots could be purchased, but the price of thousand to tens-of-thousands of dollars made them cost prohibitive to the modest budgets of smaller educational institutions. With the recent advances of inexpensive computational components, robot platforms have become readily accessible to modest budgets.

More importantly these platforms have made the area of robotics accessible to everyone by removing the need to have a complete background in electrical engineering, mechanical engineering, and computer science at the same time. Platforms such as the Handyboard and the LEGO RCX (Martin et al.2000) have managed to allow users to cross the *threshold of indignation*, which is "the maximal behavioral component that we are willing to make to get a task done" (Saffo 1996). If end users perceive that their efforts go beyond this point, no matter how good or interesting a manufacturer believes a new product to be, it will not succeed in the consumer market.

From the electrical engineering side a variety of preconstructed sensors as well as motors are available. From the mechanical engineering side robot bodies can be constructed from the simple building blocks, which include gears, axles, and hinges. And from the computer science side there are a variety of programming languages available with commands to read sensors and control motors. (Knudsen 1999) The accessibility of these platforms coupled with their flexibility to design sufficiently interesting complex systems have made a valuable pedagogical tool for a wide variety of advanced concepts (e.g., Avanzato 2000, Gaines and Balac 2000, Jadud 2000, Meeden 1996, Norstrand 2000).

A Multidisciplinary Project Action Group

Even though these new platforms can make virtually anyone a robotics engineer, without the guidance of some formal knowledge, instructors and students alike can be overwhelmed with many impractical designs. Instructors can have difficulty forming practical and meaningful learning experiences and students can have difficulty in completing assignments due to the large design spaces. Some courses overcome this by providing elements of the project that are not in the area of study. For example, giving computer science students a specific mechanical platform and/or sensor configuration (Gaines and Balac 2000). Other courses use a structured exercise approach, which provide students a number of exercises that familiarize them with the different components and the concept of each area of study (Kumar and Meeden 1998). For this approach to be effective, instructors need to have sufficient background knowledge to formulate effective learning exercises, for example, mechanically: gears and structures, electronically: sensor limitations, and programmatically: algorithmic design and multitasking.

To address this we have formed a Robotics Multidisciplinary Project Action Group (Robotics MPAG). The Robotics MPAG consists of members from across various disciplines in the School of Engineering. This includes Computer Science, Electrical & Computer Engineering, Mechanical Engineering, and Industrial Engineering. The Robotics MPAG provides a forum and basis for sharing expertise across the disciplines with the goal of helping to form learning activities in the different disciplines that are effective for students in each of the different areas. So, material can be presented at a level that students in mechanical engineering can learn enough about structured programming principles, behavior based robotic control, and multitasking in order to implement a control program. And computer science students can learn enough about sensor processing, gearing, and transformation power in order to design the physical robot structure. In essence, the Robotics MPAG is a cross-functional design team for educational experiences.

A main aspect of this approach, members of the group create project modules for courses in their respective areas that encompass concepts to be mastered in structured exercises. These modules provide a basis of concepts and technical vocabulary for design discussions amongst the members. Through these design sessions, technical concepts of one area of study are translated to materials and exercises at a level that students in the complementary areas of study can understand. Members work together to adapt and elaborate on the modules so that the content is accessible to students outside of the specific area of expertise. So, members from the computer science department help to create robot control programming modules that are accessible to members in the other disciplines who have very little programming experience. While modules developed from the mechanical engineering are of use to courses taught in computer science where the faculty and students have little or no expertise of creating mechanical structures. In this way, faculty members are sharing their areas of expertise. Prior to this effort, the individual members of the group felt that they did not possess the necessary expertise to do hands-on robotics assignments in their courses. Table 1 shows a sample of concepts emphasized in an area of study and concepts shared with other areas of study.

Area: Course	Concepts Emphasized	Concepts Shared
Computer Science: Artificial Intelligence	Embedded agents, deliberative/reactive robot control, planning, multitasking	Subsumption architecture, search strategies, multitasking, cross compiling, multiplexing
Mechanical Engineering: Mechatronics Robotics – dynamics & control	Sensor processing, logic circuits, real-time processing, actuators, analog/digital conversion, kinematics, trajectory planning	Differential motion, gearing, translation motion
Industrial Engineering: Engineering Problem Solving	Problem formulation, structural design, algorithmic design, search strategies, gearing, drive train	Problem analysis and definition, integrated system design
Electrical & Computer Engineering: Senior Project	Signal processing, robotic system design, and project management, analog/digital conversion	Signal processing, Sensor characteristics, robotic system integration

Table 1: A sample of concepts emphasized and shared

Since the formation of the Robotics Multidisciplinary Project Action Group robotics projects have been included in a freshman level Industrial Engineering course on engineering problem solving, a senior level Computer Science on artificial intelligence, a senior level Mechanical Engineering courses control systems, and senior projects courses in Computer Science and Electrical Engineering. Robotics projects are also a part of outreach programs that introduce pre-college students to science and engineering. These include a Girl Scout camp and a minority outreach program.

A Hybrid Robot Control Project

In addition to robotics projects in courses, the Robotics MPAG has also formed multidisciplinary graduate and undergraduate project committees. These committees serve to further the activity of cross-fertilization of expertise, and the projects serve as a basis for creating course material for teaching advanced concepts. A current project titled "Implementation of a Hybrid Robot Control Architecture on an Inexpensive Robot Platform" is exploring the development of a robot control program that includes both a reactive and a deliberative control component on the LEGO RCX. The intent of the project is to investigate the capability of the LEGO RCX hardware and software with respect to advanced robot-control architectures. The LEGO RCX being originally designed for the consumer market has plug-and-play sensors and motors. This makes the platform more accessible to faculty who would like to teach robot control concepts but do not have an electrical engineering background to build sensors. The goal is to fold the lessons learned into course material, and use the developed architectures as a basis for student projects.

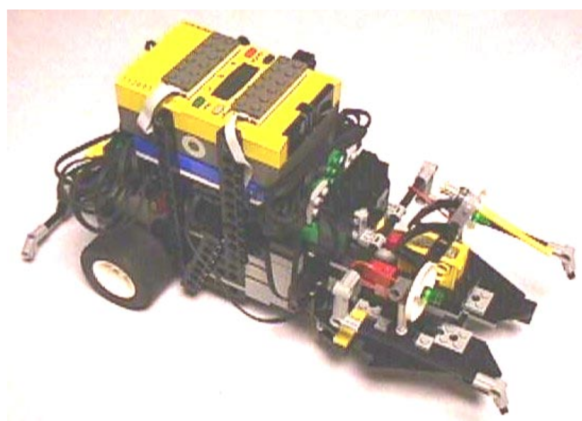

Figure 1: Foraging robot

The project uses a foraging task as its behavioral model (See Figure 1). In reactive mode, the robot wanders, avoids obstacles, and searches for targets. Once found, targets are captured. The robot then searches for its starting location ("home"), releases the target when it arrives there, and then begins looking for another target. The current setup uses a second RCX at home that emits a beacon message to guide the reactive robot in the correct direction. In the deliberative and hybrid modes the robot begins with a map of the arena. The project will investigate the robot's reactions to both known and unknown targets and obstacles. Using a purely deliberative paradigm, the robot will step through a SENSE-PLAN-ACT sequence for determining the best route to a target and again for returning home. Encounters with unknown obstacles will require re-planning prior to continuation. If unknown targets are encountered, they will be noted for future reference. With the hybrid approach, the robot will also plan the best route to a goal state (either home or target). If the plan fails, the robot will go into a reactive mode to recover. The navigation task for these architectures is accomplished using dead reckoning based distance and direction data.

The project has a number of interesting challenges presented by the constraints of the hardware, firmware, sensors, and mechanical components such as the limited number of sensor inputs. The robot uses 6 sensors: two touch sensors, two rotation sensors, a light sensor, and a compass sensor. To overcome the limitation of the LEGO RCX having only 3 sensor input ports, a multiplexor is being used[1]. The multiplexor design uses one motor port to actively switch between one of three sets of the other five ports. The sets of ports are divided between the major behavioral goals of the robot as shown in Table 2. Note that even with the multiplexor, there was still a need to place the light sensor and touch sensors on the same port. This is possible since the touch sensor is passive and provides a value of 100% when it's input is interpreted as a light sensor. As long as the values for the actual light readings do not exceed about 95% then the two sensors can be reliably configured this way. Since both touch sensors are on the same port, the robot has no way of interpreting which bumper is hit. Thus, it simply reverses the direction that it was heading.

Robot State:	**HOME**	**TRAP / Reconnoiter**	**SEARCH**
Port A	FORWARD	REVERSE	OFF
3	Rotation (Turn)	--	Rotation (Turn)
2	Rotation (Fwd/Rev)	Compass (Heading)	Rotation (Fwd/Rev)
1	Touch (Front/Rear Bumper)	Light (Target Sensor)	Touch (Front/Rear Bumper) Light (Target Sensor)
B	Motor (Turn)	Motor (Trap Target)	Motor (Turn)
C	Motor (Fwd/Rev)	--	Motor (Fwd/Rev)

Table 2: Multiplexor Port Configuration

[1] The multiplexor, called a "backpack", was designed and donated by John Barnes.

The mobile robot navigation is an interesting and challenging problem in itself. Using a rotation sensor the robot can get a reasonable sense of distance traveled, but can only get a gross sense of direction. To improve on the robot's sense of direction we have designed a compass sensor for the LEGO RCX based on a digital compass sensor. It provides 8-direction distinction, which is sufficient for the task and environment. The combined information of the three sensors provides sufficient information to perform dead reckoning calculations as well as the capability for crosscheck between the sensor inputs. However, the mechanical characteristics of the LEGO building blocks themselves make it difficult for the robot to drive in a straight line, resulting in significant drift. To compensate, the robot uses a single drive train that is driven by a single motor. Dual-differentials (see Figure 2) evenly distribute the power between two independent drive wheels. A second motor, also connected via the dual-differential drive assembly, provides power for turning in either direction. This drive configuration has helped but the remaining drift must be accounted for in computation.

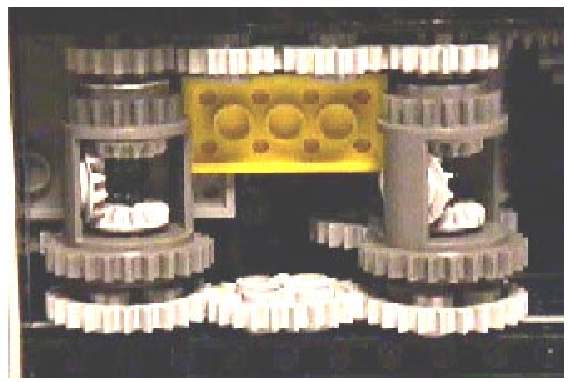

Figure 2: Dual-differential Drive

One of the largest challenges to the project is being able to develop an effective architecture using the deliberative and hybrid paradigms within the limited processor and memory capabilities of the RCX. While direct SENSE-ACT behavior-based response takes little memory and processing time, environmental data capture, storage, and processing for plan development will heavily tax the RCX resources (Murphy 2000). The current design digitizes the arena into a grid. Specifics about the grid contents are contained within an array. Due to the limited memory availability, the arena size is limited. The current project is designing to a 6 x 4 grid map. Given the size of the robot, each grid is approximately 1 square foot. To make more efficient use of the memory resources, the project is comparing the benefits of bitwise versus integer encoding. Both allow data for 2 or 3 grids to be stored within each variable. A software tool that provides for bitwise manipulation and modulus arithmetic, such as Not-Quite-C (NQC), is required.

Area of Study	Concepts
Computer Science	Reactive robot control, deliberative robot control, Hybrid robot control, dead reckoning navigation, path planning
Mechanical Engineering	Mobile robot drive train design, robot arm design
Electrical Engineering	Signal multiplexing, sensor design, signal processing

Table 3: Advanced concepts of the foraging robot

With the successful conclusion of this project we anticipate being able to create teaching modules that will allow students to explore a wide variety of advanced concepts as shown in Table 3. The successes and failures of the project are being documented to share among the members of the Robotics MPAG. This will provide the faculty some insight into the limitations of the current resources enabling them to tailor projects and guide the students as they develop solutions.

Conclusions

In any area of study the curriculum tends to narrowly focus students to that area of expertise. In the real world though complex engineered systems are created from integrating components of electrical, mechanical, and computing elements. Robotics is a medium that allows a comprehensive view of an integrated-engineered system. It affords a view of information processing from the microprocessor level of computing up through the application software. It provides a picture that illustrates the connection between the mechanical components of a system and the computing components. The multidisciplinary approach to teaching has allowed instructors across the various disciplines in the School of Engineering to present robotics projects in this integrated nature. This provides them with a rich experience to learn about complementary areas of expertise, to learn about how the different systems interact, and they to learn about design implications resulting from the interactions.

On going work of the group includes curriculum development for a cross disciplinary course on engineering design and robotics. This course would be available to students from all the areas represented by the group. A main goal of the course is to bring together students from the different areas of expertise as a cross-functional design team. This will allow students to share and learn about each other's area of expertise. Just as important, it will provide an opportunity for students to learn that different areas of expertise bring their own design biases and perspectives to a project, which allows a wider range of design ideas to be explored by the team.

Bibliography

Avanzato, R., 2000. "Mobile Robotics for Freshman Design, Research, and High School Outreach", in *Proceedings of the 2000 IEEE International Conference on System, Man & Cybernetics*, pp. 736– 739.

Gaines, D., and Balac, N. 2000. ``Using Mobile Robots to Teach Artificial Intelligence Research Skills." In *Proceedings 2000 ASEE Annual Conference*, St. Louis, June.

Jadud, M., 2000. "TeamStorms as a Theory of Instruction", in Proceedings of the 2000 IEEE International Conference on System, Man & Cybernetics, pp. 712– 717.

Knudsen, J. 1999 *The Unofficial Guide to LEGO Mindstorm Robots*, O'Reilly & Associates.

Kumar, D. and Meeden, L. 1998. " A Robot Laboratory for Teaching Artificial Intelligence" in *Proceedings of the Twenty-ninth SIGCSE Technical Symposium on Computer Science Education*, D. Joyce, ed.

Martin, F., Mikhak, B., Resnick, M., Silverman, B., and Berg, R 2000. "To Mindstorms and Beyond: Evolution of a Construction Kit for Magical Machines" in *Robots for Kids: Exploring New Technologies for Learning*; A. Druin and J. Hendler, eds., Morgan Kaufmann.

Meeden, L., 1996. "Using Robots as Introduction to Computer Science", in Proceedings of the Ninth Florida Artificial Intelligence Research Symposium, J. Stewman, ed., pp. 473-477.

Murphy, R. 2000. *Introduction to AI Robotics*, Massachusetts Institute of Technology Press.

Norstrand, B.2000. "Autonomous Robotics Projects for Learning Software Engineering", in Proceedings of the 2000 IEEE International Conference on System, Man & Cybernetics, pp. 724 – 729.

Saffo, P. 1996 "The Consumer Spectrum" in *Bringing Design to Software*, T. Winograd, ed.